Structuring 3D Medial Skeletons: A Comparative Study

Thomas Delame, Jacek Kustra, Alexandru C Telea

Vision, Modeling, and Visualization (2016)
M. Hullin, M. Stamminger, and T. Weinkauf (Eds.)

Structuring 3D Medial Skeletons: A Comparative Study

T. Delame[1] & J. Kustra[2] & A. Telea[3]

[1]Univ. Grenoble Alpes & CNRS (LJK), INRIA, [2]Philips Research, [3]Institute Johann Bernoulli, Univ. of Groningen

Abstract

Medial skeletons provide an effective alternative to boundary or volumetric representations for applications that focus on shape structure. This capability is provided by the skeletal structure, i.e.*, the curves and surfaces computed from centers of maximally inscribed balls by a process called structuration. Many several structuration methods exist, all having various challenges in terms of delivering a high-quality medial skeleton. This paper provides a first overview of existing structuration methods. We formally define the skeletal structure by giving its theoretical properties, and use these properties to propose quality criteria for structurations. We next review existing structuration methods and compare them using the established criteria. The obtained insights help both practitioners in choosing a suitable structuration method and researchers in further perfecting such methods.*

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Curve, surface, solid and object representations

1. Introduction

Medial skeletons are thin centered structures, jointly describing the topology and the geometry of a shape $\mathcal{O}$. For 3D shapes $\mathcal{O} \subset \mathbb{R}^3$, skeletons (also called medial surfaces) consist of the set of maximally inscribed balls in $\mathcal{O}$, called *medial atoms* [Blu67]. Atoms' positions and radii capture the shape geometry, while the spatial vicinity of their centers, called *medial points*, capture the shape topology. The closest points on the shape surface $\partial\mathcal{O}$ to a medial point $\mathbf{p}$ are called *feature points* of $\mathbf{p}$. Medial points can also be defined as the loci inside $\mathcal{O}$ having at least two feature points. Medial skeletons are used in many applications, *e.g.*, shape segmentation [KJT15], shape metrology [JKT13], animation [JBPS11], shape modeling [JLW10], and shape matching [SSGD03]. *Skeletonization*, the process of computing and interpreting the skeleton $\mathcal{M}$ of a shape $\mathcal{O}$, is challenging as methods have specific assumptions and limitations, which are not always evident for users [TDS*16]. Moreover, using the resulting skeletons for several of the above operations requires the *skeletal structure*, which is not explicitely computed in most *skeletonization* operations. We elaborate on this below.

Skeletal structure describes the organization of points within $\mathcal{M}$. The theoretical skeletal structure is well-understood [Mat83, Gib00, GK04], and is composed of manifolds with boundaries, called *skeletal components*, and Y-intersection curves thereof. As skeletonization produces a sampled version of $\mathcal{M}$, *i.e.,* a finite set of atoms, skeletal structure is typically approximated by a mesh representation. Constructing this mesh, a process called *structuration*, is key to being able to use medial skeletons for applications such as segmentation, shape analysis, or shape processing.

Challenge: Recent advances have made the computation of medial atoms fast, reliable, and simple [MGP10, MBC12, JKT13, JST16]. As such, we can expect increasingly many applications to use 3D skeletons. Still, without skeletal structure, using skeletons is limited mainly to applications requiring only local shape-thickness information, excluding use-cases such as shape matching, manipulation, animation, segmentation, and processing. Hence, understanding structuration is key to skeleton usefulness. Yet, structuration is rarely explicitly studied in the literature, as most works focus on medial atom computation. The quality of structurations *themselves* is rarely compared separately, and a taxonomy of such methods is lacking.

This paper covers three goals. First, we provide a state-of-the-art review of skeletal structure properties, quality criteria for structurations, and existing structuration methods (Sec. 2). Next, we qualitatively compare several such methods against these properties (Sec. 3), thereby extending recent practical skeleton-comparison efforts [SYJT13, SJT14a]. Finally, we provide a practical guideline to choose suitable structuration methods based on the outlined requirements (Sec. 4). Sec. 5 concludes the paper. Implementation [Del16] and shapes [SJT14b] used in our comparisons are freely available.

2. Medial Skeleton Structuration

2.1. Skeletal Structure Properties

The skeletal structure of a 3D shape $\mathcal{O}$ forms a 2-dimensional Whitney stratified set [Mat83]. Simply put, the skeleton $\mathcal{M}$ of $\mathcal{O}$ consists of closed smooth 2-manifolds $\mathcal{M}_2$ (surface components), $\mathcal{M}_1 \subset \mathcal{M}_2$ (curve components) and $\mathcal{M}_0 \subset \mathcal{M}_1$ (point components). $\mathcal{M}$ contains curves that are either *(i)* intersections between $\mathcal{M}_2$ components, named Y-intersection curves [Dam06, CLK09] or *junctions*; or *(ii) curve components* formed by atom centers in areas where $\mathcal{O}$ has a local symmetry axis. The union of curve components and $\mathcal{M}_2$ boundaries gives the *skeleton boundary* $\partial\mathcal{M}$. Junctions have neighborhoods homeomorphic to $n > 2$ closed half-disks glued together; skeleton borders have vicinities homeomorphic to either a closed half-disk or a closed interval; the remainder of the skeleton has vicinities homeomorphic to a closed disk. As skeletal structure is homotopic to, and captures the part-whole structure of, a shape, $\mathcal{M}_2$ and/or $\mathcal{M}_1$ components correspond to different shape parts, and junctions correspond to loci where parts meet. $\mathcal{M}_2$ boundaries that are not junctions correspond to curvature maxima, or convex edges, on $\partial\mathcal{O}$. Hence, *if* we can reliably extract skeletal structure, we can enable a wide range of shape analysis operations [LK07].

2.2. Skeletal Structure Quality

Since we cannot (efficiently) analytically represent all shapes $\mathcal{O}$, we cannot compute their exact skeletons $\mathcal{M}$ [TDS*16]. Instead, one approximates $\mathcal{O}$ by sampling in $\mathbb{O} = \textbf{Samp}(\mathcal{O})$, and computes a finite number of atoms $\textbf{Skel}(\mathbb{O})$. Skeletal structure is approximated from $\textbf{Skel}(\mathbb{O})$ by a structuration operation **R**, yielding $\mathbb{M} = \textbf{R} \circ \textbf{Skel} \circ \textbf{Samp}(\mathcal{O})$, an approximation of the exact skeleton $\mathcal{M}$. For example, if $\mathbb{O}$ is a finite union-of-balls approximating $\mathcal{O}$, $\textbf{Skel}(\mathbb{O})$ is the set of balls in $\mathbb{O}$ that are not completely included into other balls [MGP10], which can be very efficiently computed [MBC12, JKT13]. The challenge here is computing the structuration **R** so that $\mathbb{M}$ approximates $\mathcal{M}$ well. The quality of $\mathbb{M}$ is given by the quality of its medial atoms $\textbf{Skel}(\mathbb{O})$ and the quality of the structuration **R**. Assessing the former can be easily done *e.g.* by testing the above-mentioned ball-inclusion property, see *e.g.* [MBC12, JKT13]. Assessing structuration quality is much harder, as this requires comparing the structure of $\mathbb{M}$ to that of the exact skeleton $\mathcal{M}$ which is usually unavailable.

To tackle this, consider the sampled version $\textbf{Samp}(\mathcal{M})$ of the ideal skeleton $\mathcal{M}$. Ideally, our computed discrete skeleton $\mathbb{M}$ should be very similar to $\textbf{Samp}(\mathcal{M})$, *i.e.*, skeletonization and sampling should be commutative. We cannot compare these two skeletons, as we do not have $\mathcal{M}$ and thus nor $\textbf{Samp}(\mathcal{M})$. However, we can infer the theoretical structural properties of $\textbf{Samp}(\mathcal{M})$ and require these to hold also for our computed skeleton $\mathbb{M}$. If this happens, $\mathbb{M}$ is indeed, structure-wise, a good approximation of $\mathcal{M}$. We next infer the following key structural properties for $\mathbb{M}$:

- **R1:** $\mathbb{M}$ consists of triangles between atom centers (for surface components) and edges (for curve components, if any);
- **R2:** triangles and lines only intersect at shared edges and/or vertices;
- **R3:** edges shared by no triangle are in the curve components;
- **R4:** edges shared by only one triangle are on the skeletal surface-components' border;
- **R5:** edges shared by more than two triangles are at junctions;
- **R6:** $\mathbb{M}$ has the same homotopy as $\mathbb{O}$. In particular, this forbids edges between non intersecting medial balls.

A good structuration method should also guarantee convergence [SYJT13, SJT14a]: As $\mathbb{O}$ approaches $\mathcal{O}$, so should $\mathbb{M}$ approach $\mathcal{M}$. As for the above properties, we do not measure convergence directly, but by checking how structural properties get increasingly satisfied as $\mathbb{O}$ approaches $\mathcal{O}$.

All above form the *functional* properties that the output of a structuration method **R** should comply with. Additionally, a good structuration method should also satisfy several *non-functional*: Scalability, ease of use, automated parameter setting, and ability to treat shapes $\mathcal{O}$ of different types (*e.g.*, watertight or not) and different sampling qualities for $\mathbb{O}$ (*e.g.*, having sparse/non-uniform samples, different triangle aspect ratios and sizes, degenerate triangles, and inconsistently oriented triangles).

2.3. Structuration Methods

We next review several of well-known structuration methods against the quality criteria in Sec. 2.2. We group these methods in two classes: methods that use the input shape $\mathbb{O}$ (Sec. 2.3.1); and methods that use solely medial atoms $\textbf{Skel}(\mathbb{O})$ (Sec. 2.3.2). Methods in the first class are, by definition, more constrained by the input quality and less generic than methods in the second class.

2.3.1. Structuration with Input Shape

Methods in this class are typically embedded in the computation of **Skel**. This enables such methods to use all information obtained when computing **Skel** to evaluate **R**, *i.e.*, perform the structuration. A drawback is that such methods are intimately connected to assumptions and technical aspects of **Skel**, which makes them less flexible.

2.3.1.1. Voronoi Diagrams (VD): Given a sampled surface $\partial\mathbb{O}$, we can easily compute its Voronoi diagram $V(\partial\mathbb{O})$ by considering all samples as sites [AM97, CL05]. Voronoi vertices inside $\mathbb{O}$, along with their distance to the nearest site, form the finite union-of-balls approximation of $\mathbb{O}$. A VD structuration simply considers the subsets of Voronoi edges and Voronoi faces in $V(\partial\mathbb{O})$ that *(i)* connect internal Voronoi vertices and *(ii)* remain completely inside $\mathbb{O}$. This first known 3D structuration method was motivated by theoretical results guaranteeing (though in 2D only, see Sec. 2.3.1.4) the convergence $\mathbb{M}$ to $\mathcal{M}$ as $\partial\mathbb{O}$ approaches $\partial\mathcal{O}$ [BA92].

Since all Voronoi faces are planar, they can easily be divided into triangles to match **R1**. Due to the nature of a Voronoi diagram, triangles never intersect in their interiors. The number of triangles shared by an edge also matches the ideal skeletal structure. Yet, spurious triangles appear and complicate the skeletal structure – a recurring issue when computing medial atoms by a Voronoi diagram [ACK01, CL05]. Finally, triangles and edges that connect non-intersecting atoms need to be filtered so that $\mathbb{M}$ has the same homotopy as $\mathbb{O}$. This can create isolated atoms and thus lead to a different number of connected components in $\mathbb{M}$ and $\mathbb{O}$. Regarding non-functional requirements, the VD method is quite slow, as it has a complexity of $O(n^2)$ [ES92] for n samples in $\mathbb{O}$. Using a spatial divide-and-conquer approach, the computation of the Delaunay triangulation, dual to the VD, can be parallelized [JPT15]. The VD structuration can only be applied when atoms are computed by a Voronoi skeletonization. Thus, the computation time must include both the computation of atoms and the structuration. The VD method is easy to use, openly available in the CGAL library [JPT15], and parameter-free. There is no convergence guarantee, since at higher density, spurious triangles still exist. This is the main reason that motivated the creation of the Powershape method (Sec. 2.3.1.4).

2.3.1.2. Bisectors (B): Bisector methods are closely related to VD methods. The bisector $B(\mathbf{p},\mathbf{q}) \subset \mathbb{R}^3$ of a point-pair $(\mathbf{p},\mathbf{q}) \in \mathbb{O}^2$ is the set of points equidistant from $\mathbf{p}$ and $\mathbf{q}$. Bisector methods [Lee82, CKM04] compute $B(\mathbf{p},\mathbf{q})$ for all sample pairs in $\mathbb{O}$, and prune these next to yield their subsets $H(\mathbf{p},\mathbf{q}) \subset B(\mathbf{p},\mathbf{q})$ of points that are closest to $\mathbf{p}$ and $\mathbf{q}$. A non-empty $H(\mathbf{p},\mathbf{q})$ is the Voronoi face dual to the Delaunay edge $\mathbf{pq}$ of $\mathbb{O}$. Restricting $\cup H(p,q)$ to the inside of $\mathbb{O}$ yields thus the desired structuration. While similar to VD methods, bisector computation is typically far less efficient than VD construction. All bisectors are computed by brute force, and most get discarded in the pruning step. Medial scaffold methods [LK01, LK03, LK07] alleviate this by classifying medial points to compute only bisector parts relevant to the skeleton $\mathbb{M}$. The resulting atoms and skeletal structure are as accurate as those produced by VD methods. While medial scaffold methods are still slower than VD methods, they also deliver a medial point classification atop the structuration, which enables several shape analysis applications (Sec. 2.1). However, because remaining bisector parts are exactly inside Voronoi edges, the Bisector structuration does not converge in 3D case to the exact $\mathcal{M}$ structure when sampling density increase.

Given the above relation, the Bisector structuration meets functional requirements identically to VD structurations. For the non-functional requirements, the only differences are the computation time (longer for Bisectors than VD) and the ease of use (to our knowledge, there is no publicly available Bisectors implementation).

2.3.1.3. Delaunay Structuration (DS): This structuration method was introduced as a post-processing step of the shrinking ball method in [JKT13], which is a faster version of [MBC12]. Here, $\partial\mathbb{O}$ is a mesh with triangle faces $\mathcal{F} = \{F_i = \{\mathbf{f}_{ij}\} \subset \partial\mathbb{O}\}$. For each face F_i, the medial points $\mathbf{x} \in \mathbf{Skel}(\mathbb{O})$ having a vertex $\mathbf{f}_{ij}$ as feature point are collected, then projected on the plane of F_i, and triangulated by a standard 2D Delaunay triangulation. The triangulation is then projected back on the medial points to yield a local 'patch' of $\mathbb{M}$. The 2D Delaunay triangulation is very cheap, as there is a small number of feature points to project and triangulate by face F_i. Also, faces F_i can be processed in parallel, yielding a very fast structuration overall. However, this method does not guarantee any of the properties **R2**...**R6**. This is due to the fact that faces F_i of $\partial\mathbb{O}$ are processed *independently* and the well-known fact that different areas (faces) of a surface $\partial\mathbb{O}$ can share many skeleton points in convex regions of $\partial\mathbb{O}$ [TDS*16, SP09].

2.3.1.4. Power Shape (PS): Given a dense enough uniform sampling $\partial\mathbb{O}$ of a 2D shape $\partial\mathcal{O}$, a provably convergent and homotopic approximation of the exact medial skeleton $\mathcal{M}$ is achievable using VDs [BA92, AM97]. However, this does not directly hold for 3D shapes: Even for arbitrarily fine samplings, the Delaunay triangulation contains so-called *sliver tetrahedra*. Such tetrahedra correspond to VD vertices which neither fall close to the skeleton nor are related to any prominent surface feature. The PS method addresses this by only considering the so-called *Voronoi poles* [AB98]. For each site $\mathbf{p} \in \partial\mathbb{O}$, the vertices of its convex Voronoi polyhedron on the two sides of $\partial\mathbb{O}$ that are farthest from $\mathbf{p}$ are the Voronoi poles of $\mathbf{p}$. The polar balls of $\mathbf{p}$ are the balls passing by $\mathbf{p}$ centered in its Voronoi poles. The validity of Voronoi poles for medial approximation was formally verified in [ABK98]. Since Voronoi poles are a subset of Voronoi vertices, the Voronoi diagram cannot be used to reconstruct the skeletal structure, as holes would appear. Instead, a power diagram of the polar balls is constructed. A power cell is a weighted version of a Voronoi cell obtained with the power distance metric between two polar balls $B_i(\mathbf{x_i}, r_i)$ and $B_j(\mathbf{x_j}, r_j)$ defined by: $d_{pow}(B_i, B_j) = \|\mathbf{x}_i - \mathbf{x}_j\|^2 - r_i^2 - r_j^2$ and $d_{pow}(B_i, \mathbf{x} \in \mathbb{R}^3) = \|\mathbf{x}_i - \mathbf{x}\|^2 - r_i^2$. The set of power cells forms the power diagram, which is the dual of the regular triangulation of the points $\mathbf{x}_i$ weighted by the values r_i. The PS reconstruction is the set of edges and faces of regular triangulation that *(i)* connect inside Voronoi poles and *(ii)* remain completely inside $\mathbb{O}$.

The PS method is much slower than the VD methods, since PS needs to compute a (more expensive) power diagram atop the VD computation. As for the VD method, the PS method can be parallelized via a divide-and-conquer strategy. Compared to all previous methods, the PS method has a convergence guarantee: The PS structure $\mathbb{M}$ approaches the exact skeletal structure $\mathcal{M}$ when $\partial\mathbb{O}$ is an ε-sampling of $\mathcal{O}$, as ε vanishes, and $\mathcal{O}$ is C^1, which is typical for all (dense) samplings of 3D shapes [ABK98].

2.3.1.5. Collapsed Surface (CS): The collapsed surface method exploits the fact that a surface skeleton should be homotopic with its shape (Sec. 2.1). Hence, $\mathbb{M}$ is computed simply by 'collapsing' the mesh $\mathbb{O}$ to the medial atom locations $\mathbf{Skel}(\mathbb{O})$ along the feature vectors. Implementing this is much simpler than all other methods reviewed here, and yield high visual-quality medial surfaces [JT12]. Yet, none of the properties **R2**...**R5** is preserved: The resulting $\mathbb{M}$ is just a 'triangle soup' that only guarantees its location on $\mathcal{M}$. As such, $\mathbb{M}$ cannot be used for any analysis or processing operations besides visualizing the surface skeleton.

2.3.2. Methods Using Only Medial Atoms

We are aware of two methods in this class: point cloud reconstruction and weighted alpha shapes reconstruction.

2.3.2.1. Point Cloud Reconstruction (PC): PC methods are used to reconstruct 3D surfaces from unorganized point clouds describing arbitrary 3D surfaces, such as obtained by object scanning or time-of-flight cameras. Several PC methods have been used to structure surface skeletons $\mathbb{M}$, *e.g.*, isotopic reconstruction [DLRW09], tensor voting [MM07], and ball pivoting [BMR*99]. The results satisfy properties **R2**...**R6** poorly [KJT14, CLK09]. The methods in [KJT14, CLK09] are, to our knowledge, the only PC methods able to effectively handle complex medial point clouds. However, they need a very dense sampling $\mathbb{O}$ of $\partial\mathcal{O}$ to work well. On the positive side, all PC methods can be directly combined with any skeletonization method **Skel** which outputs a medial point cloud (with or without inscribed ball radii).

2.3.2.2. Weighted Alpha Shapes (AS): As oulined earlier, $\mathbb{M}$ should be homotopic to that of $\mathbb{O}$. Thus, if two medial atoms a_i and a_j are connected, their positions $\mathbf{x}_i$ and $\mathbf{x}_j$ should be close, and the atoms describe parts of $\partial\mathbb{O}$ that are adjacent. Let us consider the medial cloud $\mathbf{Skel}(\mathbb{O}) = \{a_i = (\mathbf{x_i}, r_i)\}$. Two atoms a_i and a_j describe adjacent surface parts if they intersect and if ∂a_i and ∂a_j meet in an arc that is not contained in any other ball. Another way to express this is by using the power distance d_{pow} (see Sec. 2.3.1.4): $\exists p \in \mathbb{R}^3$ such that $d_{pow}(a_i, p) = 0$, $d_{pow}(a_j, p) = 0$ and $\forall a_k \in \mathbf{Skel}(\mathbb{O}) \setminus \{a_i, a_j\}, d_{pow}(a_k, p) > 0$. Noting that a point p is a ball with a null radius, the above definition means exactly that the edge $(\mathbf{x_i}, \mathbf{x_j})$ in the regular triangulation $RT(\mathbf{Skel}(\mathbb{O}))$ is a finite non interior edge of the weighted $0-$shape [Ede92].

Given the above, a simple way to obtain the skeletal structure $\mathbb{M}$ from $\mathbf{Skel}(\mathbb{O})$ while being homotopic to the union of balls is to first compute $RT(\mathbf{Skel}(\mathbb{O}))$, then extract the finite non interior simplices of the weighted $0-$shape. This structuration method is very similar to the PS method. The main difference lies in the use of both inside and outside balls in the PS structuration. The reason is that PS was introduced to reconstruct a point cloud. The RT of all poles allows the use of an heuristic to determine which poles are inside the shape.

3. Empirical Comparison

We complement the theoretical comparison in Sec. 2 by an empirical comparison of several structuration methods.

3.1. Materials and Methods

Shapes: We chose 50 shapes, sampled as triangle meshes (in PLY format), from a recent skeletonization benchmark [SJT14b]. The shapes are of various genii, kinds (organic *vs* man-made; lumped *vs* articulated), sampling resolutions, topologies (watertight *vs* open), and level-of-detail (smooth/simple *vs* having fine surface details).

Figure 1: *Comparison of ten structurations for six shapes (see Sec. 3.2).*

Skeletonization methods: Prior to structuration, we need to compute a medial-atom set, *i.e.*, choose an implementation for **Skel** (Sec. 2.2). We consider here three such methods:

1. **Shrinking:** This is the method in [MBC12] and further accelerated in [JKT13]. The method has a simple and robust implementation, and is very fast, due to parallelization;
2. **Polar:** This is the first part of the PS method, *i.e.*, computes medial atoms as Voronoi vertices that are also Voronoi poles (Sec. 2.3.1.4). PS is one of the best known skeletonization methods in the literature;
3. **Voronoi:** This is the first part of the VD method (Sec. 2.3.1.1), *i.e.* computes medial atoms as vertices of the Voronoi diagram of $\partial\mathbb{O}$ that are inside $\partial\mathbb{O}$. We chose this method for its simplicity, precision, and widespread use.

Structuration methods: Having the medial atoms **Skel**($\mathbb{O}$), we next consider five structuration methods **R**. As methods using the input shape (Sec. 2.3.1), we consider Voronoi Diagrams (VD), Power Shape (PS), and Delaunay Structuration (DS). Bisectors (Sec. 2.3.1.2) are not considered as they are slower; have no advantage over VD for structuration; and no simple or public implementation is known to us. Collapsed Surfaces are not considered since they do not even remotely satisfy **R2**..**R6** (Sec. 2.3.1.5). As methods using only medial atoms, we consider Weighted Alpha Shapes (AS) and and the Manifold Separation (MS) method in [KJT14]. All other generic point-cloud reconstruction methods mentioned in Sec. 2.3.2 are barely suitable for medial surfaces, and thus of limited interest.

Regularization: Usually, atoms of **Skel**($\mathbb{O}$) are filtered, to remove so-called spurious branches. This process, called *regularization* [TDS*16], cannot be used here since some structurations, such as VD and PS, need the full set of atoms, otherwise they produce a very sparse meshing of the skeleton. Structurations like AS and MS can use a regularized medial atom-set. For fair comparison, and since we want to compare structuration and not regularization, we do not regularize skeletons prior to applying any of the tested structurations.

Comparison: We first visually compare structurations produced by all studied methods on all shapes (Sec. 3.2), in line with earlier works [SYJT13, SJT14a]. Next, we detect violations of the functional properties in Sec. 2.2 and illustrate these for each studied method. Finally, we assess non-functional properties – execution speed, ease of use, and automatic parameter setting (Sec. 3.3).

3.2. Results

Figures 1 and 2 show structuration results for 12 shapes of our test-set which are also popular in other skeletonization papers [MBC12, TDS*16, KJT14, MGP10, JKT13]. Top labels give the shapes' number of vertices and faces. We see that all combinations of skeletonization **Skel** and structuration **R** methods can handle shapes of widely varying geometries, topologies, and mesh sizes ranging from 23K triangles (*bird*) to almost a million (*dragon*). Open shapes like *hand* are not handled well by the Polar and Voronoi skeletonizations, so all structurations using these yield spurious long medial triangles (Fig. 2, red markers). In contrast, all structurations using Shrinking skeletonization yield skeletons bounded to the shape's convex hull (Fig. 2, green markers).

We see a large variability in the amount of noisy terminal skeleton-branches for the tested combinations of skeletonization and structuration methods and shapes: On the same model, different method combinations can yield both smooth and noisy skeletons (*cow*, *elephant*); always smooth skeletons (*bird*, *scapula*); or always noisy skeletons (*armadillo*). Among skeletonization methods, Shrinking yields less noisy structurations than Polar, which is in turn less noisy than Voronoi, regardless of the structuration method used (DS, AS, MS). Between structuration methods, MS yields the by far least noisy results for all shapes. This is explainable if we study MS in detail [KJT14]: Internally, MS performs an outlier removal step *prior* to the manifold extraction *and* aims to construct smooth, large, manifolds – something that none of the other structuration methods do.

3.3. Detailed Comparison

We refine the global insights obtained so far by a detailed study of the tested structuration methods. To assess scalability, Table. 1 gives running times for our tests, done on a 3.7 GHz PC with 32 GB RAM.

Voronoi Diagrams (VD): This method delivers skeletal structure with optimal quality for moderate or high sampling densities of $\mathbb{O}$. VD is the only structuration to guarantee **R2**...**R5** (see also Fig. 3). Hence, the produced skeletons are easily segmentable into components, see *e.g.* [LK07]. As such, the lack of theoretical convergence of $\mathbb{M}$ to $\mathcal{M}$ does not weigh much in practice. Yet, VD can only be used with a Voronoi skeletonization, which limits its usability. Also, when the sampling density of $\mathbb{O}$ is locally low, holes can appear (Fig. 4a). VD is the third-fastest skeletonization-structuration combination, being surpassed only by Shrinking with AS or DS (Table. 1). Given the many optimized and numerically accurate VD implementations available, VD is a good practical choice.

Power Shape (PS): The main issue of PS is its low speed (Table. 1) – it is it the slowest of all tested methods, regardless of the skeletonization being used. Also, PS has the limitations of both VD (holes, constrained by the skeletonization method) and AS (slow and not fulfilling properties **R2**...**R5**, see next), while having none of their advantages (high quality result for VD, independence on skeletonization and homotopy preservation for AS). In contrast, AS produces skeletal structures having similar quality (Fig. 3). Given this, we recommend to use PS only when there is no way to detect if a Voronoi vertex is inside the input shape, as it was the case in the original PS paper [ABK98].

Delaunay Structuration (DS): As stated in Sec. 2.3.1.3, and visible in Figs. 3, 5, and also in Fig. 1 (*elephant*) and 2 (*kitten*, *horse*), DS ranks lowest regarding **R2**...**R6**. This becomes clear if we study this technique [JKT13], which essentially combines purely *local* planar Delaunay triangulations. Yet, DS is by far the fastest structuration method for all types of skeletonizations used (Table. 1). Hence, DS is optimal when one requires fast skeleton computations to be used purely for visualization.

Weighted Alpha Shapes (AS): When VD is not available, AS produces the best skeletal structure (Fig. 5), while preserving, by definition, the union-of-balls homotopy. AS also allows finding balls completely included into other balls by analyzing isolated atoms. Indeed, by definition of the weighted $0-$shape, an isolated atom is either not intersecting any other atoms (easy to check) or does not describe a part of $\partial\mathbb{O}$ adjacent to parts described by other atoms, *i.e.*, not describe any part of $\partial\mathcal{O}$ at all. This can be used to remove unnecessary atoms or to enable a fast Scale Axis Transform [MGP10], without needing a costly Power Shape execution. AS can be best used as a fallback method, when VD or the input shape $\partial\mathbb{O}$ are not available, and when running time is not critical. Regarding scalability, GPU

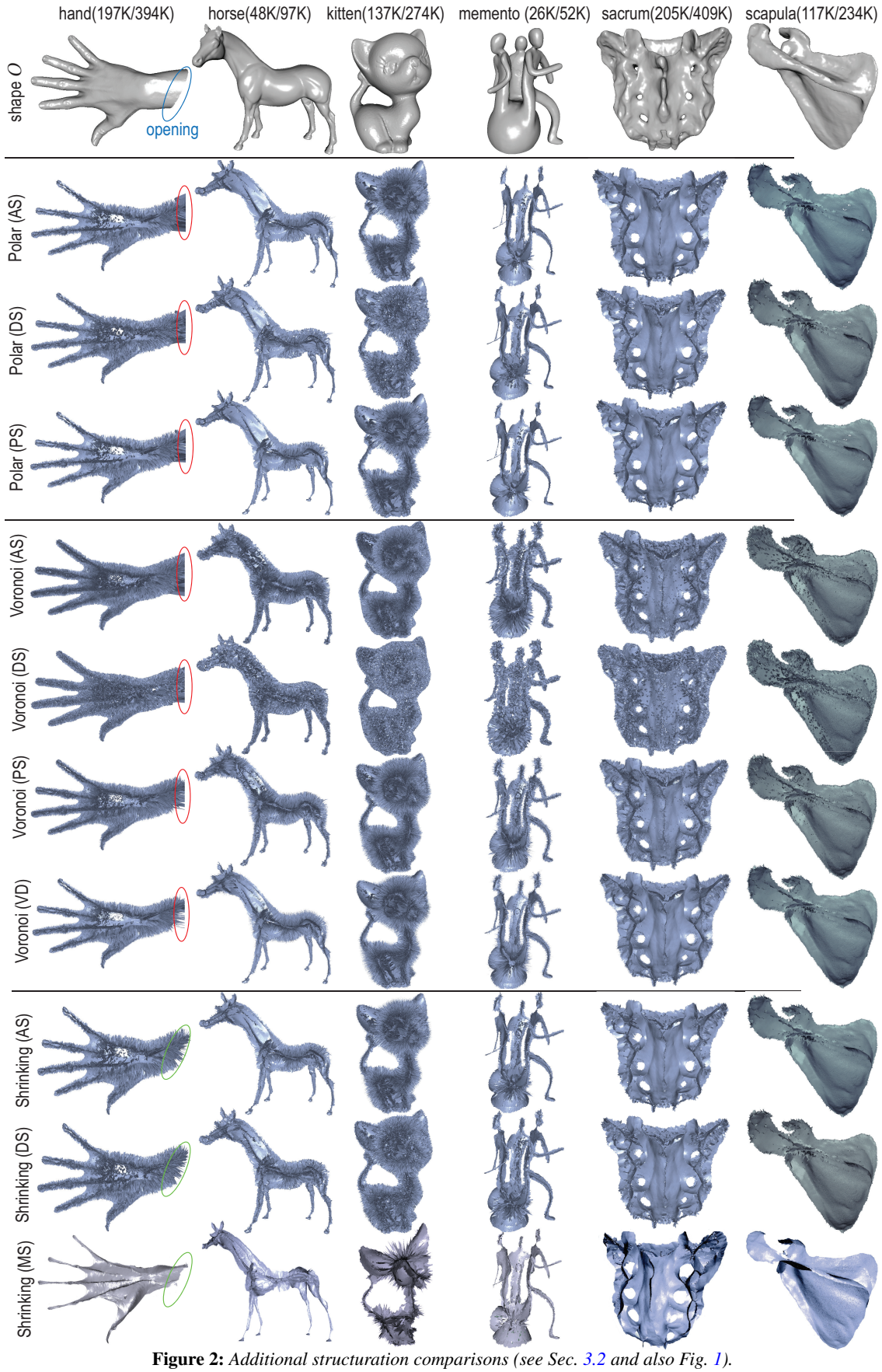

Figure 2: *Additional structuration comparisons (see Sec. 3.2 and also Fig. 1).*

Shape	Voronoi Balls Skeletonization					Polar Balls Skeletonization				Shrinking Balls Skeletonization			
	#Atoms	VD	PS	DS	AS	#Atoms	PS	DS	AS	#Atoms	DS	AS	MS
Bird	38 513	0.49	69.97	2.71	16.88	10 450	6.58	0.79	1.39	11 718	0.22	0.35	3.89
Cow	44 157	0.57	79.03	2.79	21.78	13 257	7.81	1.01	2.16	13 388	0.28	0.50	4.84
Dragon	1 557 858	23.21	2912.52	115.10	965.50	434 162	288.24	36.40	92.47	437 645	10.63	22.43	84.40
Elephant	187 263	2.48	302.86	10.52	90.68	49 936	27.79	3.46	7.27	50 485	0.86	1.98	25.51
Frog	117 105	1.79	246.73	7.79	61.10	35 265	24.10	2.92	6.14	37 225	0.79	2.44	20.49
Hand	676 886	10.72	1413.03	40.53	523.36	197 137	145.28	15.71	56.24	197 245	5.01	11.14	35.42
Horse	165 557	2.48	304.94	11.73	101.90	48 322	31.60	4.03	11.96	48 485	1.21	3.72	22.36
Kitten	462 772	7.35	1010.31	28.52	428.01	137 082	116.67	11.36	58.94	137 098	5.08	13.49	28.92
Memento	90 394	1.14	152.95	6.55	50.10	25 940	15.37	1.93	4.84	26 277	0.54	1.26	6.50
Sacrum	700 762	11.02	1329.35	53.35	357.31	204 223	136.67	18.26	32.77	204 710	5.61	10.53	36.76
Scapula	382 894	6.25	658.36	26.23	116.04	111 658	68.00	9.13	11.28	116 930	2.84	4.16	23.20

Table 1: *Execution times (in seconds) of skeletonization and structuration, and number of produced medial atoms, for the tested shapes.*

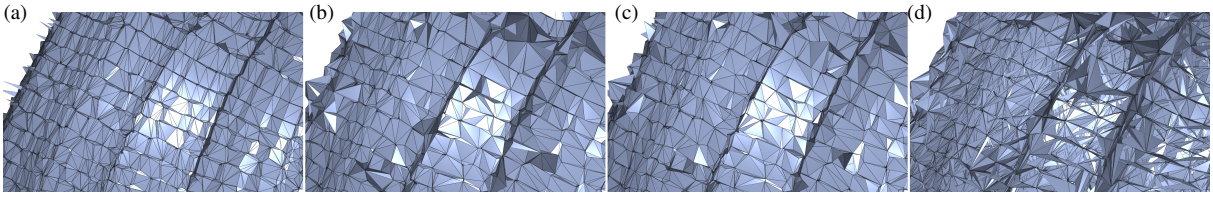

Figure 3: *Zoom-in of structurations of* frog *model using the Voronoi skeletonization. (a) VD fulfills requirements **R2** . . . **R5**. (b) AS, (c) PS and (d) DS do not respect **R2** . . . **R5**. Also, DS yields more triangle intersections and around component borders and junctions.*

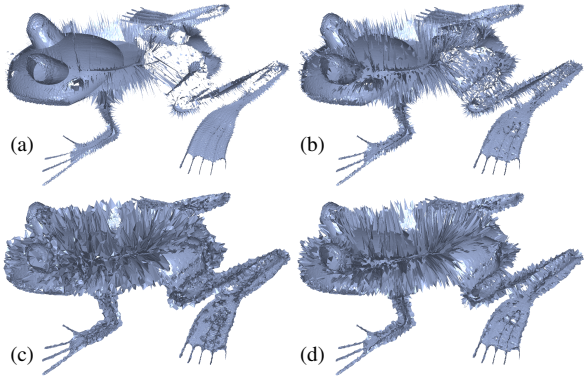

Figure 4: *Holes created by (a) VD and (b) PS for poor locally sampled $\partial\mathbb{O}$. No holes are created here by (c) DS nor (d) AS. VD and PS yield less triangles on the frog's head than AS, but yield isolated edges (not drawn here for visual clarity).*

implementations of Regular Triangulation [Nan12] could arguably make AS suitable for large high-resolution shapes.

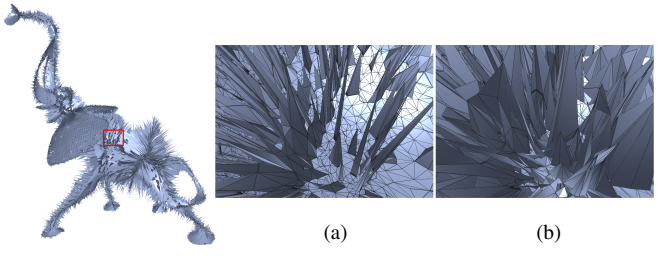

Figure 5: *Comparison between (a) AS and (b) DS around a surface component with noisy atoms. We see that AS behaves almost like VD.*

Manifold Separation (MS): Like AS, MS only requires the medial atom cloud. While MS is roughly 3 to 10 times slower than AS (being more efficient for large shapes, see Table. 1, last two columns), it also includes an outlier-removal regularization step (Sec. 3.1) which delivers *considerably* cleaner and simpler medial surfaces (Figs. 1, 2, bottom rows). MS also satisfies **R5** by construction (see [KJT14]). **R4** is not always fulfilled, as skeletal sheets are structured using the ball pivoting method [BMR*99] which can create small holes depending on its parameter settings. Our practical experience has shown that fine-tuning the several parameters of MS (noise removal, minimal manifold size, ball pivoting reconstruction) is delicate. This is in high contrast with all other studied methods, which are parameter-free.

4. Discussion

Table. 2 summarizes our analysis, including also the Bisectors and Collapsed Surface methods which we did not practically test for mentioned reasons. While no single method is optimal in all respects, several practical conclusions can be drawn, as follows.

Functional requirements: All methods guarantee **R1** by design. The only practical method to guarantee **R2**. . . **R5** is VD, which is also relatively simple to implement, but requires a dense input $\partial\mathbb{O}$ to work well. No method fulfills all **R1**. . . **R6**.

Speed: The two fastest methods (DS and CS) are also poorest in terms of functional requirements, making them only usable for visualization. AS, MS, and VD have reasonable speed, making them practical for large shapes. All methods parallelize, except B.

Ease of use: All methods are parameter-free, except MS. AS only requires the medial atom cloud, making it useful in contexts when the input shape is no longer available.

Convergence: Only AS and PS converge to the true medial surface as the sampling $\partial\mathbb{O}$ approaches $\partial\mathcal{O}$. This is useful in cases where one would be willing to sample finer to obtain a more accurate skeleton.

Overall, AS (combined with the fast Shrinking skeletonization) yields the best coverage of all studied criteria. If noise-free skeletons are mandatory, MS is best; alternatively, AS could be applied on an already regularized atom clouds.

Skeletonization **Skel** and structuration **R** are linked in subtle ways. The constraints, quality, and speed of **Skel** massively influence the final result, and not any **Skel** can be combined with any **R**. However, as shown, **Skel** and **R** can and should be studied separately to yield a way to choose between the large set of available combinations.

Structuration methods	No $\partial\mathbb{O}$ needed	Convergence	Speed	Parallelizable	Parameter free	R1	R2-R5	R6
Voronoi Diagrams (VD)	✗	✗	Average	✓	✓	✓	✓	✗
Bisectors (B)	✗	✗	Very slow	✗	✓	✓	✓	✗
Delaunay Structuration (DS)	✗	✗	Very fast	✓	✓	✓	✗	✗
Collapsed Surface (CS)	✗	✗	Very fast	✓	✓	✓	✗	✗
Power Shape (PS)	✗	✓	Slow	✓	✓	✓	✗	✗
Manifold Separation (MS)	✓	✗	Average	✓	✗	✓	✗	✗
Alpha Shapes (AS)	✓	✓	Fast	✓	✓	✓	✗	✓

Table 2: *Summary of functional and non-functional analysis of the tested structuration methods (see Sec. 4).*

5. Conclusion

In this paper, we have presented a comparative study of seven structuration methods that compute medial surfaces of 3D mesh shapes. For this, we analyzed several combinations of skeletonization and structuration techniques on a variety of shapes, and showed a total of 120 structuration results, and studied these results against six functional and five non-functional quality criteria. Our results highlight several so far not documented aspects of existing structuration methods, and allow us to propose a guideline for selecting such methods as a function of the desirable criteria to optimize for.

Future work targets a more in-depth, quantitative, study of the satisfaction of functional requirements by existing structuration methods. Separately, we believe that the insights shown here will guide and motivate researchers towards the design of better structuration methods for practical 3D skeleton computation.

References

[AB98] AMENTA N., BERN M.: Surface reconstruction by Voronoi filtering. In *Proc. ACM SCG* (1998), pp. 39–48. 3

[ABK98] AMENTA N., BERN M., KAMVYSSELIS M.: A new Voronoi-based surface reconstruction algorithm. *ACM TOG* (1998), 415–421. 3, 5

[ACK01] AMENTA N., CHOI S., KOLLURI R. K.: The power crust. In *Proc. ACM SMA* (2001), pp. 249–266. 2

[AM97] ATTALI D., MONTANVERT A.: Computing and simplifying 2D and 3D continuous skeletons. *CVIU 67*, 3 (1997), 261 – 273. 2, 3

[BA92] BRANDT J. W., ALGAZI V.: Continuous skeleton computation by voronoi diagram. *CVGIP: Imag Underst 55*, 3 (1992), 329 – 338. 2, 3

[Blu67] BLUM H.: *A transformation for extracting new descriptors of shape*. Models for the perception of speech and visual form. MIT Press, 1967. 1

[BMR*99] BERNARDINI F., MITTLEMAN J., RUSHMEIER H., SILVA C., TAUBIN G.: The ball-pivoting algorithm for surface reconstruction. *IEEE TVCG 5*, 4 (1999), 349–359. 3, 7

[CKM04] CULVER T., KEYSER J., MANOCHA D.: Exact computation of the medial axis of a polyhedron. *CAGD 21*, 1 (2004), 65–98. 2

[CL05] CHAZAL F., LIEUTIER A.: The λ-medial axis. *Graph. Models 67*, 4 (2005). 2

[CLK09] CHANG M., LEYMARIE F., KIMIA B.: Surface reconstruction from point clouds by transforming the medial scaffold. *CVIU*, 113 (2009), 1130–1146. 1, 3

[Dam06] DAMON J.: Global medial structure of regions in $\mathbb{R}^3$. *Geom. Topol. 10* (2006), 2385–2429. 1

[Del16] DELAME T.: 3d skeletonization source code, 2016. https://github.com/tdelame/median_path. 1

[DLRW09] DEY T., LI K., RAMOS E., WENGER R.: Isotopic reconstruction of surfaces with boundaries. *CGF 28*, 5 (2009), 1371–1382. 3

[Ede92] EDELSBRUNNER H.: *Weighted Alpha Shapes*. Tech. rep., Champaign, IL, USA, 1992. 3

[ES92] EDELSBRUNNER H., SHAH N. R.: Incremental topological flipping works for regular triangulations. In *Proc. ACM SCG* (1992). 2

[Gib00] GIBLIN P.: Symmetry sets and medial axes in two and three dimensions. In *Proc. IMA*. Springer, 2000, pp. 306–321. 1

[GK04] GIBLIN P., KIMIA B.: A formal classification of 3D medial axis points and their local geometry. *IEEE TPAMI* (2004), 238–251. 1

[JBPS11] JACOBSON A., BARAN I., POPOVIC J., SORKINE O.: Bounded biharmonic weights for real-time deformation. *ACM TOG 30*, 4 (2011), 78:1–78:8. 1

[JKT13] JALBA A. C., KUSTRA J., TELEA A.: Surface and curve skeletonization of large 3D models on the GPU. *IEEE TPAMI 35*, 6 (2013), 1495–1508. 1, 2, 3, 5

[JLW10] JI Z., LIU L., WANG Y.: B-mesh: A modeling system for base meshes of 3D articulated shapes. *CGF 29*, 7 (2010), 2169–2178. 1

[JPT15] JAMIN C., PION S., TEILLAUD M.: 3D triangulations. In *CGAL User and Reference Manual*, 4.7 ed. 2015. 2

[JST16] JALBA A., SOBIECKI A., TELEA A.: An unified multiscale framework for planar, surface, and curve skeletonization. *IEEE TPAMI 38*, 1 (2016), 30–45. 1

[JT12] JALBA A., TELEA A.: Computing curve skeletons from medial surfaces of 3D shapes. In *Proc. Theory & Practice of Computer Graphics (TPCG)* (2012), Eurographics, pp. 99–106. 3

[KJT14] KUSTRA J., JALBA A., TELEA A.: Robust segmentation of multiple intersecting manifolds from unoriented noisy point clouds. *CGF 33*, 1 (2014), 73–87. 3, 5, 7

[KJT15] KUSTRA J., JALBA A., TELEA A.: Computing refined skeletal features from medial point clouds. *Patt Recog Lett 76* (2015), 13–21. 1

[Lee82] LEE D.: Medial axis transformation of a planar shape. *IEEE TPAMI*, 4 (1982), 363–369. 2

[LK01] LEYMARIE F., KIMIA B.: The shock scaffold for representing 3D shape. In *Visual Form*, vol. 2059. Springer, 2001, pp. 216–227. 2

[LK03] LEYMARIE F., KIMIA B.: Computation of the shock scaffold for unorganized point clouds in 3D. In *Proc. IEEE CVPR* (2003), vol. 1, pp. 345–353. 2

[LK07] LEYMARIE F., KIMIA B.: The medial scaffold of 3D unorganized point clouds. *IEEE TVCG 29*, 2 (2007), 313–330. 1, 2, 5

[Mat83] MATHER J.: Distance from a sub-manifold in Euclidean space. In *Proc. Symp. in Pure Mathematics* (1983). 1

[MBC12] MA J., BAE S., CHOI S.: 3D medial axis point approximation using nearest neighbors and the normal field. *Visual Comput 28*, 1 (2012), 7–19. 1, 2, 3, 5

[MGP10] MIKLOS B., GIESEN J., PAULY M.: Discrete scale axis representations for 3D geometry. *ACM TOG 29*, 4 (2010), 1–10. 1, 2, 5

[MM07] MORDOHAI P., MEDIONI G.: *Tensor voting: a perceptual organization approach to computer vision and machine learning*. Morgan & Claypool, 2007. 3

[Nan12] NANJAPPA A.: *Delaunay Triangulation in R^3 on the GPU*. PhD thesis, Visvesvaraya Tech. Univ., India, 2012. 7

[SJT14a] SOBIECKI A., JALBA A., TELEA A.: Comparison of curve and surface skeletonization methods for voxel shapes. *Pattern Recogn. Lett. 47* (Oct. 2014), 147–156. 1, 2, 5

[SJT14b] SOCIECKI A., JALBA A., TELEA A.: 3D skeletonization benchmark, 2014. cs.rug.nl/svcg/SkelBenchmark. 1, 3

[SP09] SIDDIQI K., PIZER S.: *Medial Representations: Mathematics, Algorithms and Applications*. Springer, 2009. 3

[SSGD03] SUNDAR H., SILVER D., GAGVANI N., DICKINSON S.: Skeleton based shape matching and retrieval. In *Proc. SMI* (2003), pp. 130–137. 1

[SYJT13] SOBIECKI A., YASAN H., JALBA A., TELEA A.: Qualitative comparison of contraction-based curve skeletonization methods. In *Proc. ISMM* (2013), Springer, pp. 425–439. 1, 2, 5

[TDS*16] TAGLIASACCHI A., DELAME T., SPAGNUOLO M., AMENTA N., TELEA A.: 3D skeletons: A state-of-the-art report. *CGF 35*, 2 (2016), 573–597. 1, 2, 3, 5